IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A SURVEY ON KEY-AGGREGATE SEARCHABLE ENCRYPTION FOR GROUP DATA SHARING IN CLOUD STORAGE

Swapnil D. Raut*, Prof. Avinash P. Wadhe, Prof. Jayant P. Mehare
Dept. Of Computer Science and Engineering, Sant Gagde Baba Amravati University G. H. Raisoni College Of Engineering And Management

DOI: 10.5281/zenodo.48345

ABSTRACT

Security concerns over inadvertent data leaks in the cloud may greatly ease the capability of selectively sharing encrypted data with different users via public cloud storage. So designing such an encryption schemes is a key challenge which lies in the efficient management of encryption keys. When any group of selected documents need to share with any group of users a desired flexibility is required with demands different encryption keys, which are used for different documents. However this also indicates the need of securely sharing to users a large number of keys for encryption and search, and those users will have to safely save the received keys, and submit an equally large number of keywords trapdoors to the cloud in order to perform search over the shared data. The indicated purpose of safe communication, storage, and difficultly clearly renders the approach impractical. In this paper, we address this practical problem, which is greatly neglected in the literature, here we are proposing the new concept of key aggregate searchable encryption and instantiating the concept through a concrete KASE scheme. In this scheme, the documents are shared by just submitting a single trapdoor by the user to the cloud for querying and this single key is being received by the data owner for sharing large number of documents. Our proposed scheme can confirm prove both the safety as well as practically efficient channels by security analysis and performance evaluation. It can securely store and manage the users in their devices. In order to perform a keyword search over many files a large number of trapdoors must be generated by users and submitted to the cloud. Such a system with secure communication, storage and computational complexity may lead to inefficiency and impracticality.

KEYWORDS: Searchable encryption, data sharing, cloud storage, data privacy.

INTRODUCTION

Cloud Storage has been promising solution to provide straighter, convenient and on demand access to a large amount of data shared on Internet sharing photos, videos and personal data through social media applications on cloud are vivid instances. Business houses also infatuated towards cloud storage which consist lower cost, agility and optimum resource use.

The leakage of data in the cloud, however, mars the conveniences of data sharing's. Such unauthorised data leaks by a malicious adversary or a misbehaving cloud operator generally results in a serious assault on personal privacy or business secrets.(e.g recently a high profiled celebrity photos have been leaked in icloud).To ensure user's security over potential data leaks in cloud storage, the data owner has to encrypt all the data before uploading it on the cloud and later such data can be retrieved and decrypted by those who have decryption keys. Such a cloud storage is called as cryptographic cloud storage. The encryption of data forbids the users to search and retrieval is made only after providing given keys. One obvious remedy is to devise a searchable encryption scheme(SE) along with potential keywords by the data owner before uploading them on the cloud. Keyword should be matched to retrieve data, the user has to send the corresponding keyword trapdoor to the cloud for access over the encrypted data.

Through a searchable encryption keys are combined with cryptographic cloud storage to meet basic security needs, to implement such a system for large scale applications with millions of users and billions of files are hindered by practical issues of effective management of encryption keys. To share encrypted data with different users, the prerequisite is different encryption keys for the data access. The data owner should distribute a number of keys to users to search the encrypted files and to decrypt the files. Such a large number of keys not only to be distributed to users via secured channels but also to be wisely stored and managed by the users in their devices. Apart from this, trapdoors are needed to be generated and submitted to the cloud while searching many files.

This paper puts forward the novel concept of key aggregate searchable (KASE),and initiating the concept through concrete KASE scheme. The proposed KASE scheme is applicable to any cloud storage, that support searchable group data sharing functionality to enable any user selectively share a group of selected files with a group of selected users. The prerequisites of efficient key management are twofold. First, a data owner needs to distribute a single aggregate key to a user for sharing any file. Secondly, the user needs to submit a single aggregate trapdoor (instead of group of trapdoors) to the cloud to perform keyword search over any number of shared files. The proposed KASE scheme in this paper meets the need of key-aggregate cryptosystem which inspires this paper.

Whet I am going to contribute through this paper?
1] I put forward here a general framework of key aggregate searchable encryption(KASE) which is composed of seven polynomial algorithms for security parameter setup, key generation, encryption, key extraction ,trapdoor generation, trapdoor adjustment and trapdoor testing. This has been followed by description of both functional and security requirements for designing a valid KASE scheme.
2] Another feature of this paper is formulation of KASE framework by designing a concrete KASE scheme. There is through analysis of the efficiency of the scheme that establishes it's security.
3] It deals with practical issues while building an actual group data sharing system based on the proposed KASE system to evaluate it's performance. The evaluation stresses the practically of the KASE scheme.
The rest of the paper follows the following steps.
1] I have conducted through review of background knowledge.
2] General KASE framework has been defined.
3] Review of related work has been given.
4] Designing a concrete KASE Scheme and its analysis emphasizing efficiency and security.
5] Practically and evaluation of KASE based group data sharing system.
6] The last section comprises conclusion.

LITERATURE REVIEW

This section places forward reviews of several categories of solutions that are in existence and these work brigs out their relevance.

1.1 Multi user searchable Encryption

A rich literature has been available on searchable encryption. Including SSE schemes [5]-[8] and PEKS schemes [9]-[15]. Contradictory to those existing work, in the control of cloud storage keyword search under the multi tenancy setting is a more common scenario. In such a scenario, the data owner would like to share a document with a group of authorized users and the user who has access right can provide a trapdoor to perform the keyword search over the shared document namely the "Multi user searchable encryption" (MUSE) scenario.

Some recent work [6],[13]-[15], [19] focus to such a MUSE scenario. Though they all adopt single key combined with access control to achieve the goal. In [6],[19], Muse scheme are constructed by sharing the document's searchable encryption key with all users who can access its and broadcast encrypting is used to achieve coarse joined access control. In [13]-[18], attributes based encryption (ABE) is applied to achieve line grained access control aware keyword search. The main problem in MUSE has been to control users who can access documents, In order to reduce the number of shared keys and trapdoors are not considered. Key aggregate searchable encryption can provide the solution for the latter and it can make MUSE more efficient and practical.

1.2 Multi key Searchable Encryptions

In multi user application the number of trapdoors are proportional to the number of documents to search over different provides to the server a keyword trapdoor under each key which have to be matched and document can be encrypted Popa[28] firstly introduces the concept of multi key searchable encryption (MKSE) and places forward the first feasible scheme in 2013 MUSE enables a user to provide a single keyword trapdoor to the server, but still allows the server to search for that trapdoor's keyword in documents encrypted with different keys. KASE is altogether different from MKSE.

KASE delegates the keyword search right to any user by distributing the aggregate key to him/her in a group data sharing system while the goal of MKSE is to ensure the cloud server can perform keyword search with one trapdoor over different documents owing to a user.

1.3 Key Aggregate Encryption for Data Sharing

Data sharing system based on closed storage has much priority now days [1]-[4]. In particular, Chu et al. [4] consider how to reduce the number of distributed data encryption keys sharing different document with different encryption keys with the same user the data owner will need to distribute all such keys to him/her in a traditional approach which is usually impractical. In order to resolve this problem key aggregate encryption (KAE) scheme for data sharing is proposed to generate an aggregate key for the user to decrypt all the documents.

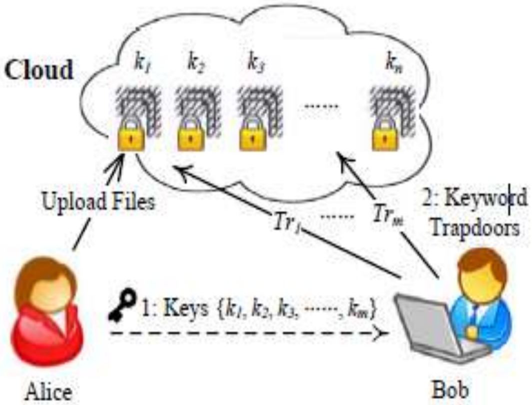
A set of documents encrypted by different keys to be decrypted with a single aggregate key so that user can encrypt a message both under a public key and under the identifier of each documents The construction is inspired by the broadcast encryption key [27] The data owner can be regarded as the broadcaster who has public key pk and master key msk Every document with identifier's can be regarded as a receiver listening to the broadcast channel and a public information used in decryption is designed to be relevant to both the owner's msk and the encryption key the message encryption process has resemblance with data encryption using symmetric encryption in BE but the key aggregation and data encryption are regarded as mathematical transformation of BR Encrypt algorithm and BE Decrypt algorithm respectively.

PROPOSED WORK

Lets consider that two employee of a company are sharing same confidential business data using public cloud storage system(e.g dropbox or syncplicity).For instance, Emily wants to upload a large collection of financial documents to the cloud storage that are meant for directors of different departments to review. These documents have high sensitive information which should only be accessed by authorised users. Jonathan is one of the directors who has been authorized to view documents concerning his department. Emily encrypts these documents with different keys and generates keyword cipher texts based on department names before uploading to the cloud storage. Emily much delegate to Jonathan the rights for keyword search over these documents and decryption of documents related to Jonathan's department.

Traditionally, Emily must securely send all the searchable encryption key to Jonatahn. After receiving these keys, Jonathan must store them securely. He must also generate all the keyword trapdoors using these keys in order to perform a keyword search. As is shown in fig.1(a) Emily is assumed to have private documents set{doci}i=1n. And for each doci ,a searchable encryption key ki is used. Without loss of generality ,suppose Emily wants to share m documents{doci} $\frac{m}{i}$ = 1 with Jonathan. In this case, Emily must sent all the searchable encryption keys{Ki}im to Jonathan when Jonathan wants to retrieve documents containing a keyword w, he must generate keyword trapdoor. Tr, for each document doc; with key ki and submit all trapdoors{Tri}im to the cloud server, when m is sufficiently large, both key word distribution, storage and trapdoor generation can become too expensive for Jonathan's client-side device, which basically defies the purpose of using cloud storage.

This paper places forward the novel consent of key aggregate searchable encryption (KASE) as a better solution as depicted in fig.1(b) in KASE, Emily only needs it distribute a single aggregate key instead of (k)im =1 for sharing in document with Jonathan, and Jonathan only needs to submit a single aggregate trapdoor instead of {Tri}im =1 to the cloud server. The cloud server can use this aggregate trapdoor and some public information to



(a) Traditional approach

keyword search in group data sharing system.

Fig. 1

perform keyword search and return the result to Jonathan. Therefore, in KASE, the delegation of keyword search right can be achieved by sharing the single aggregate key. However, it remains an open problem to delegate the keyword search rights along with the decryption rights. Which has been the center of interest of this paper. The problem of formulating a KASE scheme can be stated as –

"To design a key, aggregate searchable encryption scheme under which any subset of keyword cipher text (produced by the SE encrypt algorithm to be introduced in section S from any set of documents is searchable (performed by the SE Test algorithm) with a constant size trapdoor (produced by SE Trapdoor algorithm) generated by a constant size aggregate key".

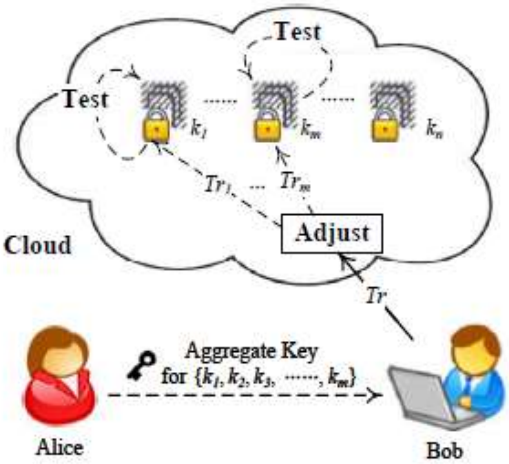


Fig 1.(B)

This paper places forward the novel consent of key aggregate searchable encryption (KASE) as a better solution as depicted in (i.g. /c6) in KASE, Emily only needs it distribute a single aggregate key instead of (k);m =1 for sharing in document with Jonathan, and Jonathan only needs to submit a single aggregate trapdoor instead of {Tri}im =1 to the cloud server. The cloud server can use this aggregate trapdoor and some public information to perform keyword search and return the result to Jonathan. Therefore, in KASE, the delegation of keyword search right can be achieved by sharing the single aggregate key. However, it remains an open problem to delegate the keyword search rights along with the decryption rights. Which has been the center of interest of this paper. The problem of formulating a KASE scheme can be stated as –

"To design a key, aggregate searchable encryption scheme under which any subset of keyword cipher text (produced by the SE encrypt algorithm to be introduced in section S from any set of documents is searchable (performed by the SE Test algorithm) with a constant size trapdoor (produced by SE Trapdoor algorithm) generated by a constant size aggregate key".

3.2 The KASE FRAMEWORK

The KASE FRAMEWORK comprises seven algorithms. In order to set up the scheme, the cloud server would generate public parameters of the system through the setup algorithms and these public parameters can be opted again and again by different data owners to share their files. For each data owner, he/she should produce a public/ master secret key pair through keygen algorithm. Keywords of each document can be encrypted via the Encrypt algorithm with the unique searchable encryption key. After this, the data owner will use the master secret key to generate an aggregate searchable encryption key for a group of selected documents via the extract algorithm. The aggregate keys are distributed securely to authorized users who need to access Lose documents. An authorized user has to produce keyword trapdoor via the Trapdoor algorithm using this aggregate key and submit the trapdoor to the cloud as is illustrated in fig. after receiving the trapdoor to perform the keyword search over the specified set of documents, the cloud server will run the Adjust algorithm to generate the right trapdoor for each document and them run the Test algorithm to test whether the document contains the keyword this frame
Work can be summarized as—

- Setup
- Keygen
- Encrypt
- Extract
- Trapdoor
- Adjust
- Test

3.3 Requirements for designing KASE Schemes:

The KASE Framework discussed in the previous section highlights the designing of KASE Scheme. But a valid KASE scheme must fulfill several functional and security parameters.
As stated below—

- Compactness A KASE scheme has to ensure the size of aggregate key that should be independent of the number of files to be shared. In order to set keys{KI}iϵs'. it requires that key←Extract(MSK,S) How to aggregate the set of keys into a single key without invalidating later steps in a key challenge in designing KASE schemes.
- Searchability if a focal point to all KASE schemes as it generates desired trapdoors to any given keyword for searching encrypted documents. Reduction of the number of keys should preserved the search capability. Methodically for each document containing the keyword to with index iϵs. the searchability requires that iϵ(tr=Trapdoor (Keys,w)) and Tri←Adjust(params,I,s,Tr),Then Test (Tri,i)=True.
- Delegation to delegate the key search right to a user thought as aggregate key has been the main goal of KASE. The data owner has to assign the delegated key that can perform keyword search, in which the inmates of the adjustment algorithm must not be public i.e. it should not rely or any user's private information. This has been the second challenge while designing KASE scheme. Apart from this, KASE scheme should also satisfy two security tenets as under--
- Controlled searching. The attaches can't search for an arbitrary word without authorization of the data owner. The attaches cannot perform keyword search over the documents that are not relevant to the known aggregate key. He or she can't generate new aggregate searchable encryption key for the set of document from the recognized keys.

Quarry privacy. It means the attaches cannot determine the keyword used in a quarry, away from the information that can be acquired observation and the information derived from it. The user can ask the untreated cloud server to search for a sensitive word without revealing the word to the server.

CONCLUSIONS

Having considered various issues privacy preservation of data sharing system based on public cloud storage in which the data owner has to distribute a large number of keys to users to impart accessibility over the documents, this paper proposes the novel concept of key aggregate searchable encryption (KASE) and construct a concrete KASA scheme. Analysis and evaluation results justify that this work can provide en effective solution to building practical data sharing system based on public cloud storage.

In a kasa scheme the owner only needs to distribute a "singlekey to a user when sharing a variety of documents with the user and the user only needs to submit"single trapdoor when he makes a guest over all documents shared by the same owner. If a user wants to query over documents shared by multiple owners, he must generate multiple trapdoor to the cloud. In order to reduce the number of trapdoors under multi owners setting is a future work. Federated Clouds have more attraction these days but this KASA can't be applied in this case directly.
It will be future work to provide the solution for KASA subjected to federated clouds.

REFERENCES

[1] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing" ,Proc. IEEE INFOCOM, pp. 534-542, 2010.
[2] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing", Proc. ACM Symp. Information, Computer and Comm.Security, pp. 282-292, 2010.
[3] X. Liu, Y. Zhang, B. Wang, and J. Yan. "Mona: secure multiowner data sharing for dynamic groups in the cloud", IEEE Transactions on Parallel and Distributed Systems, 2013, 24(6): 1182-1191.
[4] C. Chu, S. Chow,W. Tzeng, et al. "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 468-477.
[5] X. Song, D.Wagner, A. Perrig. "Practical techniques for searches on encrypted data", IEEE Symposium on Security and Privacy,IEEE Press, pp. 44C55, 2000.
[6] R. Curtmola, J. Garay, S. Kamara, R. Ostrovsky. "Searchable symmetric encryption: improved definitions and efficient constructions", In: Proceedings of the 13th ACM conference on Computer and Communications Security, ACM Press, pp. 79-88, 2006.
[7] P. Van,S. Sedghi, JM. Doumen. "Computationally efficient searchable symmetric encryption", Secure Data Management, pp. 87-100, 2010.
[8] S. Kamara, C. Papamanthou, T. Roeder. "Dynamic searchable symmetric encryption", Proceedings of the 2012 ACM conference on Computer and communications security (CCS), ACM, pp. 965- 976, 2012.
[9] D. Boneh, C. G, R. Ostrovsky, G. Persiano. "Public Key Encryption with Keyword Search", EUROCRYPT 2004, pp. 506C522,2004.
[10] Y. Hwang, P. Lee. "Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-user System", In: Pairing-Based Cryptography C Pairing 2007, LNCS, pp. 2-22, 2007.
[11] J. Li, Q. Wang, C. Wang. "Fuzzy keyword search over encrypted data in cloud computing", Proc. IEEE INFOCOM, pp. 1-5,2010.
[12] C. Bosch, R. Brinkma, P. Hartel. "Conjunctive wildcard search over encrypted data", Secure Data Management. LNCS, pp. 114- 127, 2011.
[13] C. Dong, G. Russello, N. Dulay. "Shared and searchable encrypted data for untrusted servers", Journal of Computer Security, pp. 367-397, 2011.
[14] F. Zhao, T. Nishide, K. Sakurai. Multi-User Keyword Search Scheme for Secure Data Sharing with Fine-Grained Access Control. Information Security and Cryptology, LNCS, pp. 406-418, 2012.
[15] J. W. Li, J. Li, X. F. Chen, et al. "Efficient Keyword Search over Encrypted Data with Fine-Grained Access Control in Hybrid Cloud", In: Network and System Security 2012, LNCS, pp. 490- 502, 2012.
[16] J. Li, K. Kim. "Hidden attribute-based signatures without anonymity revocation", Information Sciences, 180(9): 1681-1689, Elsevier, 2010.

[17] X.F. Chen, J. Li, X.Y. Huang, J.W. Li, Y. Xiang. “Secure Outsourced Attribute-based Signatures”, IEEE Trans. on Parallel and Distributed Systems, DOI.ieeecomputersociety.org/10.1109/TPDS.2013.180, 2013.

[18] J.Li, X.F. Chen, M.Q. Li, J.W. Li, P. Lee, Wenjing Lou. “Secure Deduplication with Efficient and Reliable Convergent Key Management”, IEEE Transactions on Parallel and Distributed Systems, 25(6): 1615-1625, 2014.

[19] Z. Liu, Z. Wang, X. Cheng, et al. “Multi-user Searchable Cloud”, Fourth International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), IEEE, pp. 249-255, 2013.

[20] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing”, Proc. IEEE INFOCOM, pp. 525-533, 2010.

[21] B. Wang, B. Li, and H. Li, “Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud”, Proc. 10th Int’l Conf. Applied Cryptography and Network Security, pp. 507- 525, 2012.

[22] D. Boneh, C. Gentry, B. Waters.“ Collusion resistant broadcast encryption with short ciphertexts and private keys”, Advances in CryptologyCCRYPTO 2005, pp. 258-275, 2005.

[23] D. H. Phan, D. Pointcheval, S. F. Shahandashti, et al. “Adaptive CCA broadcast encryption with constant-size secret keys and ciphertexts”, International journal of information security, 12(4): 251-265, 2013.

[24] D. Boneh, B. Lynn, H. Shacham. “Short signatures from the Weil pairing”, Advances in Cryptology ASIACRYPT 2001, pp. 514- 532, 2001.

[25] L. B. Oliveira, D. F. Aranha, E. Morais, et al. “Tinytate: Computing the tate pairing in resource-constrained sensor nodes”, IEEE Sixth IEEE International Symposium on Network Computing and Applications, pp. 318-323, 2007.

[26] M. Li, W. Lou, K. Ren. “Data security and privacy in wireless body area networks”, Wireless Communications, IEEE, 17(1): 51- 58, 2010.

[27] D. Boneh, C. Gentry and B. Waters. “Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys”, CRYPTO’05, pp. 258C275, 2005.

[28] R. A. Popa ,N. Zeldovich. “Multi-key searchable encryption”. Cryptology ePrint Archive, Report 2013/508, 2013.